REMARKS

The Examiner is thanked for his careful and very thorough Office Action.

Claims 1, 3-5, and 7-35 are pending. Claims 1, 3-5, and 7-35 have been rejected.

Note that the amendments to Claim 1 are intended to be purely formal amendments, and are believed not to change the scope of this claim. Entry of the amendments is respectfully requested.

Claims 36-44 have been added. The support for the new claims can be found, for example, in paragraphs [0012] and [0023] to [0027]. The added claims are respectfully asserted not to introduce new matter, and their entry is respectfully requested.

Art Rejections

The art rejections are all respectfully traversed.

Rejection Under 35 USC 103(a)

Claims 1, 3-5, and 7-35 stand rejected under 35 USC Section 103(a) as being unpatentable over *Baldwin* (U.S. Patent No. 6,025,853) in view of *Brent et al.* (U.S. Patent No. 5,459,864).

CLAIM 1

A graphics processor, comprising:

a plurality of parallellized graphics computational units; and

one or more task allocation units programmed to bypass defective ones of said graphics computational units within said plurality, and to distribute incoming tasks only among operative ones of said graphics computational units.

1. The asserted combination of references does not teach or suggest each limitation of Claim 1.

Specifically, Claim 1 recites, "one or more task allocation units programmed to bypass defective ones of said graphics computational units within said plurality, and to distribute incoming tasks only among operative ones of said graphics computational units."

The present innovations do not dynamically load balance as that term is used in *Brent et al.*

With regard to this limitation, the Examiner has suggested that, "It is old and well known and well used in the art to dynamically load balanced among multiple processors include skip or bypass defective unit(s)."

Applicant would like to respond to this suggestion by first pointing out that the present innovations do not **dynamically** load balance as that term is used in *Brent et al.* Dynamic load balancing, as the term is used in *Brent et al.*, gives *Brent et al.* a:

> dynamic reconfiguration feature available for use at any time to allow any processor to be removed, or a new processor to be added, to the subsystem, as long as there is at least one operational processor remaining in the subsystem. Reconfiguration involves a re-assignment of the QEs to the queues and their processors remaining after the reconfiguration. (col. 6, *ll.* 11-17).

The present innovations do not imply or claim such dynamic reconfiguration or "load balancing" as that term is used in *Brent et al.* As established, for example, by col. 6, *ll.* 11-17 of *Brent et al.* cited above, **bypassing a defective unit is not the same thing as dynamic reconfiguration or "load balancing" among processors.**

The present innovations, for example, in claim 1, claim one or more task allocation units that would allow defective units on a die to be bypassed, so that what would have been a defective die can be salvaged and used as a lower quality, lower performance graphics processor chip. This functionality

is not taught or suggested by *Brent et al.*, which teaches dynamic reconfiguration of multiple processors when one of those processors is busy or becomes inactive for some reason. As stated in *Brent et al.*:

> If a queue-associated processor is busy when its queue receives an ADM request, this invention provides ADM workload balancing by having the busy processor transfer the waiting ADM request to the queue of any other processor which is not busy and that processor then immediately performs the ADM request. When all queue-associated processors are busy, and any queue receives an ADM request, the associated processor pauses between subunits of work to move the ADM request to a common queue (CQ). The CQ is serviced by all queue-associated processors when each processor completes a request and looks for a new work request to perform. The processor looks at the CQ header to find if the CQ is empty; if the CQ is not empty, that processor will immediately move a waiting ADM request from the CQ to that processor's queue, from which it may be executed. (col. 5, *ll.* 1-15).

In contrast, claim 1 claims parallelized graphics computational units. I/O processors are not the same as graphics computational units. Also, defective units are bypassed. There is no "load balancing" as the term is used by *Brent et al.*, and there is no dynamic reconfiguration of the graphics computational units.

A graphics computational unit is not the same thing as an I/O processor.

The Examiner goes on to further suggest that:

> Furthermore, Brent teaches a load balancing, error recovery and reconfiguration control in a data movement subsystem with cooperating plural queue **processors** (Fig. 2, col. 2, lines 39-45, col. 5, lines 49-52 and col. 6, lines 11-18). It would have been obvious to one of ordinary skill in the art at the time the present invention was made to combine the teachings of bypass defective **unit** and distribute load from defective **unit** to

> other **units** of Brend into the system of Baldwin in order to automatic load balancing among plural **processors**, automatic recovery from any failing **processor**, and automatic configuration for the subsystem containing the **processors** without intervention from the operating system as taught by Brend (col. 1, lines 18-24).

With regard to this suggestion, the Examiner appears to be using the terms "processors" and "units" interchangeably. However, Applicant would like to respectfully point out that the two terms are indeed not interchangeable and that a graphics computational unit is not the same thing as an I/O processor. Accordingly, data movement within interconnected and cooperating queue processors is not the same thing as distributing incoming tasks among parallelized operative graphics computational units.

Therefore, even if one were motivated to make the suggested combination (which Applicant strongly disputes), it still would not result in the innovations of Claim 1.

2. ***Brent et al.* does not teach the problem or its source.**

As correctly noted by the Examiner in the section cited above, "Brent teaches a load balancing, error recovery and reconfiguration control in a data movement subsystem with cooperating plural queue **processors**." It does not teach the problem or source of improving the yield enhancement of complex, integrated circuit chips. As stated in paragraphs [0011] to [0016] of the present application:

> The only practical way to design high performance graphics chips involves replicating part of the design (i.e. multiple texture pipes) so that multiple operations can be carried out in parallel. As the size of the silicon die is large the expected yield from manufacturing is less than desirable. During testing a single bit error anywhere on the die will force that die to be scrapped. The technique of redundancy has been used, particularly in memory chips, whereby extra spare rows of memory cells (or some other element) are designed in and this can be used to replace a row which has an error in it.

In latest generation graphics chips the replicated parts are too big to have any spares so we do not have the notion of redundancy. However, by internally reconfiguring the chip we can still make use of a die with one or more failing texture pipes, for example. This allows us to take die which would otherwise be classified as scrap and use them in a lower performance product. To allow this to be implemented we need to have reconfigurability in the architecture, independent scan chains so that a manufacturing fault can be isolated and a method of recording the test result inside the chip.

This idea can be applied not only to complex chips in the field of graphics, but to any market where large parts of the design are replicated to give scalable performance.

The disclosed innovations, in various embodiments, provide one or more of at least the following advantages:

The ability to use partially defective die as fully functioning parts with lower performance, which are marketed and sold at a lower price point.

A reconfigurable chip design wherein performance can be scaled by changing the number of configured units.

Therefore, because *Brent et al.* does not teach the problem or source addressed by the present innovations, one of ordinary skill in the art would not be motivated to combine or modify the references in the manner required to form the solution disclosed by the present application.

3. The presently claimed inventions may be reached only through an improper use of the disclosed inventions as a template to modify the prior art to reach the claimed inventions.

In his Response to Arguments section, the Examiner correctly notes that, "Brent teaches a plurality of processors (**not graphics computational units**) where load balancing, error recovery and reconfiguration control (such as, addition or deletion includes remove/skip/bypass of defective **processor.**" (emphasis added by Applicant). The Examiner goes on to suggest that *Brent et al.* is being used, "to show that the concept of bypass defective unit in plurality of parallel units/processors and redistribute the load to other units is old and well known as proved by Brent."

With regard to this suggestion, Applicant would like to point out again that *Brent et al.* does not teach parallelized units. Rather, *Brent et al.* teaches interconnected and cooperating processors. Parallelized units are not the same thing as interconnected and cooperating queue processors. Therefore, even if one were motivated to make the suggested combination (which Applicant strongly disputes), it still would not result in the innovations of Claim 1.

More importantly, the Examiner may not make modifications to the prior art using the claimed invention as a model for the modifications. *In re Fritch*, 972 F.2d 1260, 23 U.S.P.Q.2d 1780, 1783-1784 (Fed. Cir. 1992). "The mere fact that the prior art may be modified in the manner suggested by the Examiner does not make the modification obvious unless the prior art has suggested the desirability of the modification." *Id.* In other words, unless some teaching exists in the prior art for the suggested modification, merely asserting that such a modification would be obvious to one of ordinary skill in the art is improper and cannot be used to meet the burden of establishing a *prima facie* case of obviousness. Such reliance is an impermissible use of hindsight with the benefit of Applicant's disclosure.

Accordingly, for the reasons stated above, Applicant respectfully submits that Claim 1 is not obvious in view of the asserted combination.

CLAIM 12

A method of 3D graphics rendering, comprising the actions of:

providing a plurality of parallellized graphics computational units;

bypassing defective ones of said units, and

distributing incoming tasks only among operative ones of said units.

Claim 12 also recites features not taught by the asserted combination. Specifically, Claim 12 recites, "bypassing defective ones of said units, and distributing incoming tasks only among operative ones of said units."

As stated above, *Brent et al.* does not disclose or suggest bypassing defective graphics computational units and distributing incoming tasks only

among operative graphics computational units. Data movement within interconnected and cooperating queue processors is not the same thing as distributing incoming tasks among parallelized operative graphics computational units.

Therefore, for the reasons stated above, Applicant respectfully submits that Claim 12 is not obvious in view of the asserted combination.

CLAIM 20

A computer graphics system comprising:

means for providing a plurality of parallellized graphics computational units;

means for bypassing defective ones of said units, and

means for distributing incoming tasks only among operative ones of said units.

Claim 20 also recites features not taught or suggested by the asserted combination. Specifically, Claim 20 recites, "means for bypassing defective ones of said units, and means for distributing incoming tasks only among operative ones of said units."

Again, as stated above, *Brent et al.* does not disclose or suggest a means for bypassing defective graphics computational units and distributing incoming tasks only among operative graphics computational units. Data movement within interconnected and cooperating queue processors is not the same thing as distributing incoming tasks among parallelized operative graphics computational units.

Therefore, for the reasons stated above, Applicant respectfully submits that Claim 20 is not obvious in view of the asserted combination.

CLAIM 28

A method for computer graphics system operation, comprising the actions of:

providing a plurality of parallellized rendering units;

bypassing defective ones of said units, and

distributing incoming tasks only among operative ones of said units.

Claim 28 also recites features not taught by the asserted combination. Specifically, Claim 28 recites, "bypassing defective ones of said units, and distributing incoming tasks only among operative ones of said units."

Again, as stated above, *Brent et al.* does not disclose or suggest bypassing defective graphics computational units and distributing incoming tasks only among operative graphics computational units. Data movement within interconnected and cooperating queue processors is not the same thing as distributing incoming tasks among parallelized operative graphics computational units.

Therefore, for the reasons stated above, Applicant respectfully submits that Claim 28 is not obvious in view of the asserted combination.

CLAIMS 3-5, 7-11, 13-19, 21-27, and 29-35

Dependent Claims 3-5, 7-11, 13-19, 21-27, and 29-35 depend directly or indirectly from independent Claims 1, 12, 20, and 28 and incorporate all the limitations thereof.

Thus, for the reasons discussed above, Applicant respectfully requests withdrawal of this rejection.

Conclusion

This amendment is being submitted in response to the Final Office Action dated 08/05/2004 and, therefore, could not have been submitted earlier. Its entry is respectfully requested. All grounds of rejection and/or objection are traversed or accommodated, and favorable reconsideration and allowance are respectfully requested. The Examiner is requested to telephone the undersigned attorney or Patrick C. R. Holmes for an interview to resolve any remaining issues.

Respectfully submitted,

N. Elizabeth Pham, Reg.No. 49,042

Customer Number 29106

Attorney for Applicant

Groover & Holmes
One Galleria Tower, Suite 1370
13355 Noel Road
Dallas TX 75240
972-980-5840, FAX -5841

5 OCTOBER 2004